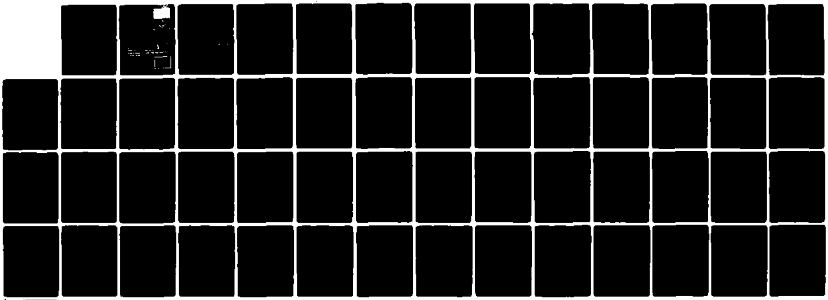
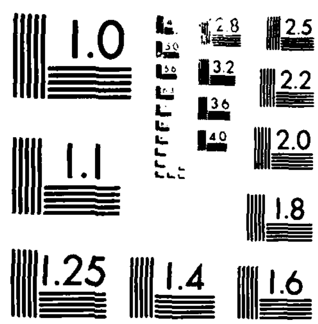
AD-A126 309 SPACEFILLING CURVES AND ROUTING PROBLEMS IN THE PLANE (U) GEORGIA INST OF TECH ATLANTA PRODUCTION AND DISTRIBUTION RESEARCH CENTER L K PLATZMAN ET AL. 1/1

UNCLASSIFIED 11 FEB 83 PDRC-83-02 N00014-80-K-0709 F/G 12/1 NL



END
DATE
FILMED
4-83
DTIC



MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS 1963 A

Georgia Institute of Technology

SPACEFILLING CURVES AND ROUTING PROBLEMS IN THE PLANE

LOREN K. PLATZMAN

JOHN J. BARTHOLDI III

PDRC 83-02

PRODUCTION and DISTRIBUTION RESEARCH CENTER

SCHOOL OF INDUSTRIAL AND SYSTEMS ENGINEERING
GEORGIA INSTITUTE OF TECHNOLOGY
A UNIT OF THE UNIVERSITY SYSTEM OF GEORGIA
ATLANTA, GEORGIA 30332

PDRC Report Series 83-02

February 11, 1983

Contract N00014-80-K-0709

SPACEFILLING CURVES AND ROUTING PROBLEMS IN THE PLANE

LOREN K. PLATZMAN

JOHN J. BARTHOLDI III

PDRC 83-02

School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta GA 30332. Telephone (404)-894-2300

SUBMITTED TO MATHEMATICS OF OPERATIONS RESEARCH

This document was prepared on a word processing system devised by Loren Platzman. It runs on the Radio Shack Model III with a very cheap printer. Please let us know what you think of it!

DISCLAIMER NOTICE

THIS DOCUMENT IS BEST QUALITY PRACTICABLE. THE COPY FURNISHED TO DTIC CONTAINED A SIGNIFICANT NUMBER OF PAGES WHICH DO NOT REPRODUCE LEGIBLY.

SPACEFILLING CURVES AND ROUTING PROBLEMS IN THE PLANE

Loren K. Platzman and John J. Bartholdi III

Georgia Institute of Technology

This paper introduces a novel heuristic to compute a minimal-length tour of N given points in the plane: they are sequenced as they appear along a spacefilling curve. The algorithm consists essentially of sorting: it is easily coded, requires only O(N) memory, and may be implemented to execute in O(N log N) operations at most, or O(N) operations on the average. If the points lie in a square of area A, the heuristic tour will have length $2\sqrt{NA}$ at most. If the points are statistically independent under a smooth distribution, with N large, then the tour will be approximately 25% longer than optimum, and a simple enhancement reduces this to 15%. We also give performance bounds for our method when the given points lie in a general subset of d-space, with an arbitrary distance measure.

AMS 1980 subject classification. Primary: 90C42. Secondary: 90B35.

OR/MS Index 1980 subject classification. Primary: 491 Networks/graphs/travelling salesman. Secondary: 632 Programming/integer/algorithms/heuristic.

Key words: Planar travelling salesman problem, heuristic algorithm, spacefilling curve.

The authors have been supported by the New Faculty Research Development Program of the Georgia Institute of Technology, and by the Office of Naval Research under Contract No. N00014-80-k-0709. Reproduction in whole or in part is permitted for any purpose of the U.S. Government.

1. Introduction. The travelling salesman problem (TSP) is to construct a circuit of minimum total length that visits each of N given points. Even in the plane, this problem is NP-complete [12]. Karp [10] has given an $O(N \log N)$ heuristic to compute a tour whose length is within ϵ of optimal, but it is difficult to code, and its effort has a constant factor that increases rapidly as ϵ decreases. Bentley and Saxe [8] devised an $O(N^{3/2} \log N)$ implementation of the nearest neighbor heuristic, but it requires a special data structure. This paper describes a faster, simpler heuristic that performs comparably. We gave a brief account of the ideas underlying our method in [3]. Here we present the algorithm in detail and analyze its performance.

Let U be a set (e.g., the unit square) within which tours are to be constructed, and define

$$C = \{\theta \mid 0 \leq \theta \leq 1\}. \tag{1.1}$$

When U has dimension 2 or more, a continuous mapping ψ from C onto U is known as a spacefilling curve. Such curves were first devised by Peano and Hilbert in the 1890's to resolve topological existence questions (cf. Hobson [9, pp 451-458]). More recently, they have attracted attention as entertaining examples of recursively defined computational procedures (cf. Aleph Naught[2], Abelson and diSessa[1, pp 94-102]). We turn them to decidedly practical use.

Let $\psi(0) = \psi(1)$. Then $\psi(\theta)$ traces out a "tour" of all the points in U as θ varies from 0 to 1. (For this reason, we find it convenient to view each $\theta \in C$ as a point on the unit circle, θ clockwise revolutions from a fixed reference point.) Given N points in U to be visited, our strategy is to sequence them as they appear along the spacefilling curve. Thus, we propose to construct tours in the following manner:

<u>SPACEFILLING HEURISTIC</u>

1) For each point ξ to be visited, compute a θ such that $\xi = \psi(\theta)$.
2) Sort the points by their corresponding θ's.

In Section 2, we analyze the performance of this heuristic when the region U, the distance measure by which tours are evaluated, and the spacefilling curve ψ are all arbitrary. Then, in Section 3, we define a particular spacefilling curve over the unit square, and show how its inverse is computed. This provides a specific algorithm to solve TSP's in the plane; its performance is discussed in Section 4. Section 5 shows how the ideas of Section 4 can be extended to more general TSP problems. Sections 6-7 contain lengthy proofs deferred from earlier sections. Concluding remarks are given in Section 8.

<u>2. General performance analysis.</u> Suppose that U lies in d-space and let $\|\cdot\|$ be a norm that determines the distance between points in U. Our analysis is based on the following three technical conditions, which, as we shall see, hold for most problems of practical interest:

(P1) The inverse of ψ is easily evaluated. Specifically, if $\xi \in U$ is a d-vector whose components each have k-bit representations, then there is a θ satisfying $\xi=\psi(\theta)$ and having an O(dk)-bit representation which may be computed in O(dk) operations. Although there may be many such θ, we must compute only one.

(P2) There is a finite constant Ω such that

$$\|\psi(\theta)-\psi(\theta')\| \leq f(|\theta-\theta'|), \quad \theta,\theta' \in C, \tag{2.1}$$

where

$$f(\Delta) = \Omega\,[\min(\Delta,1-\Delta)]^{1/d}, \qquad \Delta \in C, \tag{2.2}$$

(P3) ψ is Lebesgue measure preserving. That is, if I is an interval in C, the set $\{\ \psi(\theta)\ |\ \theta \in I\ \}$ has d-volume (area if d=2) equal to the length of I.

Remarks. (P1) ensures the efficiency of our algorithm; (P2) enables us to study its performance; and (P3) normalizes the problem and eliminates pathological cases. Intuitively, (P1) $\Rightarrow \psi$ is surjective and (P2) $\Rightarrow \psi$ is continuous, so together they imply that ψ is a spacefilling curve. Also, (P2) $\Rightarrow \psi(0) = \psi(1)$. (P3) requires that U and C be scaled so that U has unit d-volume and ψ "homogeneously fills" U.

Computational Effort.

The spacefilling heuristic requires O(dkN) operations to compute the θ's and O(N log N) operations to perform the sort, that is, O(N log N) operations in all. (To be more precise, dkN is the number of bits required to specify the problem, and the N numbers to be sorted each have O(dk)-bit representations, so the problem is solved in O(dkN log N) bit operations.) Alternatively, a randomized sorting algorithm requiring *O(N)* expected comparisons might be used (e.g. BINSORT, see Knuth[11] and Weide[15]). A heuristic tour is then obtained in O(dkN) expected operations.

This procedure also requires surprisingly little memory, only O(dN), linear in the size of the problem.

Worst-case Analysis.

Given N sorted points $\psi(\theta_1)\cdots\psi(\theta_N)$, the tour length is bounded above by $\sum_{i=1}^{N} f(\Delta_i)$, where $\Delta_i=\theta_{i+1}-\theta_i$, $i=1,\cdots,N-1$ and $\Delta_N=1+\theta_1-\theta_N$. Since this expression is concave and symmetric in $\Delta_1\cdots\Delta_N$ and since $\sum_{i=1}^{N} \Delta_i = 1$, it achieves a maximum of $Nf(1/N)$. Thus, we have proved

Theorem 2.1. The spacefilling heuristic produces a tour whose length, L, satisfies

$$L \leq Nf(1/N) = \Omega\, N^{(d-1)/d}.$$

We also consider how long the heuristic tour may be in relation to the optimal tour.

Theorem 2.2. If, for a given set of N points, the heuristic tour has length L and the optimal tour has length L^*, then

$$L/L^* \leq O(\log N).$$

The proof of Theorem 2.2 is lengthy and will be given in Section 6. We note that the dominant term in the exact bound for L/L^* (Eq. (6.5)) increases slowly in N but rapidly in d.

Probabilistic Analysis.

Suppose that the points $\xi_1, \ldots, \xi_N$ are independent random variables uniformly distributed over U. Now the heuristic tour length and the optimal tour length may be viewed as random variables. Since Ψ is measure preserving, each θ_i satisfying $\xi_i = \Psi(\theta_i)$ is uniquely determined with probability one, and is uniformly distributed over C. When N is large, the θ_i's approximate a Poisson process on C.

Define random variables

$$\rho_N^* = N^{-(d-1)/d} L^*, \; N=1,2,\ldots,$$

where L^* is the length of an optimal tour joining N random points. Beardwood, Halton and Hammersley[5] have shown that, when the Euclidean norm is used to evaluate tours, there is a β^* such that $\lim_{N\to\infty}(\rho_N^*) = \beta^*$, almost surely. Steele [14] has extended this to the stronger statement of convergence

$$\sum_{N=1}^{\infty} \mathrm{Prob}[|\rho_N^* - \beta^*|>\epsilon] < \infty, \qquad \forall\, \epsilon>0.$$

We now establish a similar result for the heuristic tour. Define random variables

$$\rho_N = N^{-(d-1)/d} L, \; N=1,2,\ldots, \tag{2.3}$$

where L is the heuristic tour length for N random points, and let

$$\mu_N = E[\rho_N]. \tag{2.4}$$

The sequence μ_N need not converge as $N \to \infty$. By Theorem 2.1, however, it is bounded:

$$0 \leq \mu_N \leq \Omega. \qquad (2.5)$$

These bounds may be tightened slightly by noting that (i) the optimal tour grows as $N^{(d-1)/d}\ \beta^*$, so

$$\beta^* \leq \lim\ \inf_N\ (\mu_N), \qquad (2.6)$$

and (ii) the increments in θ along a heuristic tour are nearly exponentially distributed with mean $1/N$, so, by concavity of $f(\cdot)$,

$$\begin{aligned} \lim\ \sup_N\ (\mu_N) & \\ & \leq N^{-(d-1)/d}\ N \int f(x)\ Ne^{-Nx}\ dx \\ & = \Omega\ \Gamma((d-1)/d). \qquad (2.7) \end{aligned}$$

Therefore, for large N, the expected heuristic tour length will be within a constant factor $\Omega\ \Gamma((d-1)/d)\ /\ \beta^*$ of the expected optimal tour length.

The following theorem will permit us to make a stronger statement regarding the ratio of heuristic to optimal tour lengths.

Theorem 2.3. Let β_N be a (deterministic) sequence such that $\beta_N - \mu_N \to 0$. Then

$$\sum_{N=1}^{\infty} \mathrm{Prob}[\, |\rho_N - \beta_N| > \epsilon] < \infty, \qquad \forall\, \epsilon > 0.$$

Proof: We slightly modify Steele's proof [14] of complete convergence of ρ_N^*. Simply redefine the random variables d_N in [14] to reflect the fact that our heuristic tour joins a point with another whose inverse image under ψ is closest in C under metric $f(\cdot)$:

$$d_N = \min\{f(|\theta_i - \theta_1|) \mid i=2,\ldots,N\}.$$

Since the θ_i's are independent and uniformly distributed on C, Eq. (2.2) of [14] may be replaced by

$$\mathrm{Prob}[d_N > t] \le [1-2f^{-1}(t)]^{N-1} \le [1-2(t/\Omega)^d]^{N-1}, \quad 0 \le t \le f(1/2).$$

and the remainder of [14] establishes the desired result. □

Since $\rho_N - \mu_N$ and $\rho_N^* - \beta^*$ both converge completely to zero as $N \to \infty$, the difference between ρ_N/ρ_N^* (the random ratio of heuristic to optimal tour lengths) and μ_N/β^* (the deterministic ratio of expected heuristic tour length to expected optimal tour length) vanishes almost surely as $N \to \infty$. Thus the spacefilling heuristic will reliably construct tours whose length is a fixed constant factor μ_N/β_N^* larger than optimal. This factor depends on the region U, the curve ψ, the distance measure $\|\cdot\|$, and (in principle although not in our experience) the problem size N. We can estimate μ_N by Monte-Carlo simulation: generate many random problems and solve them by the spacefilling heuristic. Good approximations to β^* are given in [5] for problems in subsets of d-space under the Euclidean norm. For d=2, $\beta^* \cong .765$ [6].

An additional performance measure concerns the length of the longest link along a tour. In general tours, it may be close to the diameter of U. This can lea to difficulties in certain applications, and has motivated the formulation of the bottleneck TSP problem [7]: to construct a tour whose largest link has minimal length. Our heuristic produces tours whose longest link is not much greater than the average link length, as we now show.

Theorem 2.4. The longest link along a spacefilling heuristic tour has expected length $O(\log(N)/N)^{1/d}$. Specifically, if $\hat{D}$ is the length of the longest link,

$$\limsup_{N\to\infty} E((\ln(N)/N)^{-1/d}\,\hat{D}) \leq \Omega.$$

Proof: Let $\hat{A}$ be the largest increment in θ. The increments in θ may be considered as independent exponential random variables of mean $1/N$, so the probability that none of the N increments exceeds t is $(1-\exp(-Nt))^N$. Let $t=\ln(N/s)/N$. The probability that all N increments are bounded by t becomes $(1-s/N)^N$. As N increases this becomes $\exp(-s)$. It follows that $\hat{A}$ is distributed as $\ln(N)/N - \ln(s)/N$, where s is an exponential random variable of mean 1. So as $N\to\infty$, $E(\hat{A}/[\ln(N)/N]) \to 1$. Since f is concave, $E(\hat{D}) \leq E(f(\hat{A})) \leq f(E(\hat{A}))$. □

<u>3. A Spacefilling Curve in the Unit Square.</u> We now construct a curve that fills the unit square,

$$S = \{ (x,y) \mid 0 \le x \le 1,\ 0 \le y \le 1 \}. \tag{3.1}$$

It resembles Sierpinski's curve [13] and enables our heuristic to perform especially well. It is recursively defined by breaking S into four identical subsquares, and filling each with a spacefilling curve rotated so that the four subcurves link to form a circuit (Figure 1).

We may arbitrarily specify the "starting point" of the curve to be (0,0), and the "direction" of the curve to be clockwise. Then, as θ increases from 0 to 1, the four subsquares are visited in the sequence

$$\begin{aligned}
S_0 &= \{ (x,y) \in S \mid x \le .5,\ y \le .5 \} \\
S_1 &= \{ (x,y) \in S \mid x \le .5,\ y \ge .5 \} \\
S_2 &= \{ (x,y) \in S \mid x \ge .5,\ y \ge .5 \} \\
S_3 &= \{ (x,y) \in S \mid x \ge .5,\ y \le .5 \}.
\end{aligned} \tag{3.2}$$

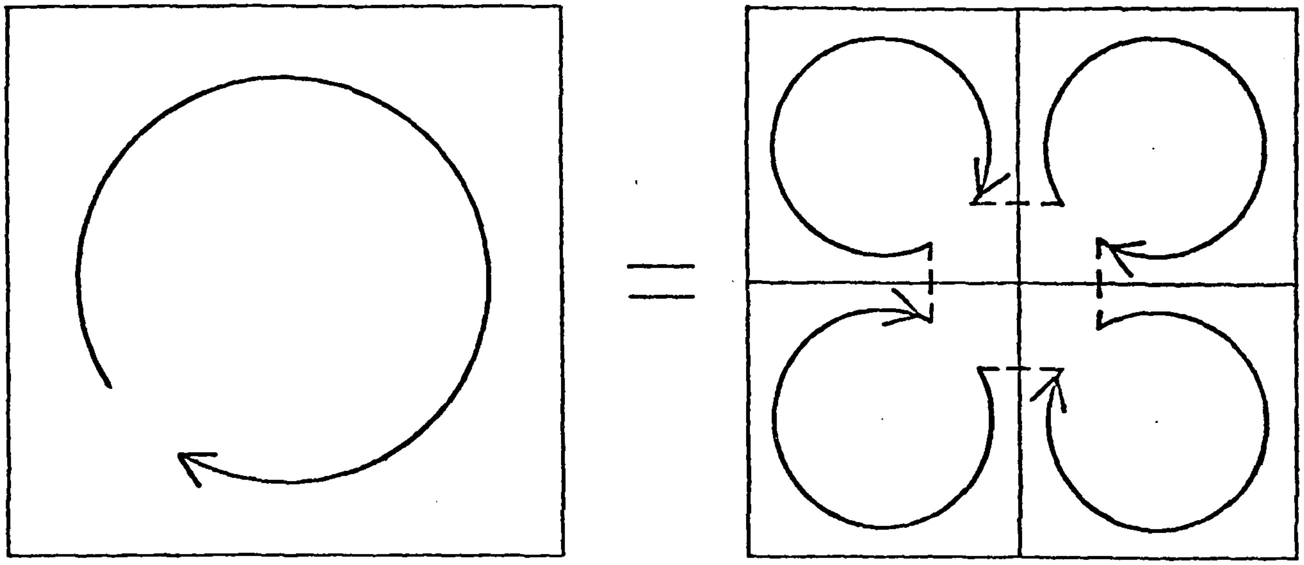


Figure 1. Structure of the spacefilling curve in the unit square.

These subsquares induce a partition of C into

$$
\begin{aligned}
C_0 &= \{\ \theta \mid 0 \le \theta \le 1/8 \text{ or } 7/8 \le \theta < 1\ \} \\
C_1 &= \{\ \theta \mid 1/8 \le \theta \le 3/8\ \} \\
C_2 &= \{\ \theta \mid 3/8 \le \theta \le 5/8\ \} \\
C_3 &= \{\ \theta \mid 5/8 \le \theta \le 7/8\ \}.
\end{aligned}
\tag{3.3}
$$

so that ψ maps C_i onto S_i, $i = 0,1,2,3$. For ease of definition we need an operation representing the rotation of each subsquare. Let R_i be an affine transformation from S to S_i determined by

$$
\begin{aligned}
R_0 &: (0,0) \to (.5,.5),\ (0,1) \to (.5,0), \\
&\qquad (1,1) \to (0,0),\ (1,0) \to (0,.5) \\
R_1 &: (0,0) \to (.5,.5),\ (0,1) \to (0,.5), \\
&\qquad (1,1) \to (0,1),\ (1,0) \to (.5,1) \\
R_2 &: (0,0) \to (.5,.5),\ (0,1) \to (.5,1), \\
&\qquad (1,1) \to (1,1),\ (1,0) \to (1,.5) \\
R_3 &: (0,0) \to (.5,.5),\ (0,1) \to (1,.5), \\
&\qquad (1,1) \to (1,0),\ (1,0) \to (.5,0).
\end{aligned}
\tag{3.4}
$$

These are interpreted as follows: position (x,y) relative to subsquare i corresponds to position $R_i(x,y)$ relative to the larger square S. For example, the point $(0,0)$ marks the start

of the spacefilling curve, so $R_i(0,0)$ is the starting point of the subcurve over S_i, that is, the center of S, (.5,.5). Finally, define affine transformations from each C_i onto C:

$$\begin{aligned}
q_0 &: 7/8 \to 0,\ 1 \to .5;\ 0 \to .5,\ 1/8 \to 1 \\
q_1 &: 1/8 \to 0,\ 1/4 \to .5,\ 3/8 \to 1 \\
q_2 &: 3/8 \to 0,\ 1/2 \to .5,\ 1/8 \to 1 \\
q_3 &: 5/8 \to 0,\ 3/4 \to .5,\ 7/8 \to 1.
\end{aligned} \tag{3.5}$$

The purpose of q_i is to convert positions along the larger curve (over S) into positions along the subcurve over S_i. For example, S_1 is covered by Ψ as θ ranges over C_1, and $\theta=.3$ is encountered $q_1(.3)=.7$ of the way through the subcurve over S_1. Using this notation, the recursive structure of Ψ, shown in Figure 1, is concisely expressed as

$$\Psi(\theta) = R_i[\Psi(q_i[\theta])], \qquad \theta \in C_i, \qquad i=0,1,2,3. \tag{3.6}$$

Figure 1 suggests that a heuristic tour will consist essentially of four subtours, each involving a fourth as many points in a fourth as large an area. Consequently, we anticipate that the expected heuristic tour length for 4N random points in S will be twice as long as that for N points. That is, $\mu_{4N} \sim \mu_N$, see (2.3) and (2.4). Yet sequences of the form $(\mu_{N \cdot 4^k})_{k=1}^{\infty}$ might approach a limit $\beta(\cdot)$ that depends on N, or, more specifically, on $4^{\log_4(N) \bmod 1}$. Such is the case, as we shall see.

It follows from (3.6) that ψ, if it exists, satisfies

$$\psi : 0 \to (0,0),\ 1/4 \to (0,1),$$
$$1/2 \to (1,1),\ 3/4 \to (1,0). \qquad (3.7)$$

Eqs. (3.6) and (3.7) provide the basis for the following algorithm to compute the inverse of ψ:

```
FUNCTION THETA(X,Y)
    1. Choose I (not necessarily unique) so that (X,Y)∈S_I
    2. If X∈(0,1) and Y∈(0,1)
          then RETURN(I/4)
          otherwise RETURN(q_I^-1[THETA(R_I^-1[X,Y])]).
```

The following theorem summarizes properties of ψ that may reasonably be inferred from our discussion thus far. Its formal proof is lengthy, however, and will be given in Section 8.

<u>Theorem 3.1.</u> The function ψ defined by (3.6) exists uniquely and is spacefilling on S. It satisfies (P1) (its inverse being computed by the function THETA), (P2) with $\|\cdot\|$ = Euclidean distance in (2.1) and $\Omega = 2$ in (2.2), and (P3). Moreover, there is a continuous function β on [1,4) such that

$$\lim_{N\to\infty} (\mu_N - \beta(4^{\log_4(N) \bmod 1})) = 0,$$

when μ_N corresponds to tours in S based on the curve defined by (3.6).

4. Performance assessment for problems in the square.

To obtain a heuristic tour in a square of area A, we rescale the points so that they lie in the unit square, compute corresponding θ values by the procedure THETA of section 3, and sort. (A short BASIC code for this algorithm was given in [3].) The tour obtained is, of course, $\sqrt{A}$ times longer than a tour in the unit square. We now compare its performance with that of the nearest neighbor heuristic (NNH) and the minimal spanning tree heuristic (MSTH), as analyzed by Bentley and Saxe [6].

Computational effort.

The worst-case effort to generate spacefilling heuristic tours is $O(N \log N)$, considerably less than those for the NNH and MSTH: $O(N^{3/2} \log N)$ and $O(N^2 \log N)$, respectively. The average effort for the NNH using a special data structure, is $O(N^{3/2})$, but the average effort for the spacefilling heuristic (using BINSORT) is $O(N)$. The computer code for the spacefilling heuristic is also simpler, and requires less memory.

Worst-case performance.

Both the spacefilling heuristic and the NNH produce tours whose length is at most $O(\sqrt{NA})$. So, as noted in [6], the ratio of heuristic to optimal tour lengths will be large only when the optimal tour is unusually small. The worst case of this ratio is $O(\log N)$ for the spacefilling heuristic with general U and ψ. For problems in the square, however, we

suspect that it is bounded by a constant. The worst instance we have been able to discover is 4.707, for a 32-point problem. A simpler example is formed by 2^k points whose θ's are the integer multiples of 2^{-k}; the ratio of heuristic to optimal tour lengths for these points is 4. The worst-case ratio for the NNH is known to lie between O(log N / log log N) and O(log N); for the MSTH, it is 2.

<u>Average performance.</u>

The expected length of a heuristic tour of N points is N times the expected length of a single link along that tour. We generated 10,000 random links for each of various values of N. From these experiments, we concluded that β(·) is nearly constant and (probably) lies in the range .956 ± .05, so that the heuristic tour is ~.956 $\sqrt{NA}$. Thus our heuristic produces tours ~25% over optimum. A simple enhancement, to be given in Section 5, reduces this to 15% over optimum. The tour obtained by the MSTH is ~.95 $\sqrt{NA}$, (~25% over optimum), and that obtained by the NNH is ~.92 $\sqrt{NA}$, (~20% over optimum).

Our experiments showed that, for tours produced by the spacefilling heuristic, the longest link is typically between 1.1 $\sqrt{(A/N)\ln N}$ and 1.3 $\sqrt{(A/N)\ln N}$. The constant preceding the radical in these expressions does not converge as $N \to \infty$. Following arguments given in [7], it can be shown that the minimum possible value of the longest link has expectation bounded below by $\sqrt{(A/\pi N)\ln N}$. Thus our method provides a reasonable solution (within an average factor of 2.3) to the bottleneck TSP. Stated differently, the expected ratio of longest link to average link along a space-filling heuristic tour is $O(\sqrt{\ln N})$. The length of the longest link along a NNH tour is comparable to the side of the containing square, so the expected ratio of longest link to average link along a NNH tour is $O(\sqrt{N})$. (This ratio is significant in applications where a single tour is partitioned into subtours for each of K vehicles. We would like subtours of equal length to contain equal numbers of customers.)

5. Generalizations. The spacefilling heuristic, as described in Section 2, is not restricted to the methods of Sections 3 and 4. Some alternatives are listed below.

Different probability distributions.

If the points are not uniformly distributed over the square, but are independent with identical probability densities $\eta(x,y)$ (and N is large), then the $\sqrt{N}$ term in the expressions for both the expected heuristic tour length and the optimal tour length is replaced by $\int_{x,y} \eta^{1/2}(x,y)dy\,dx$ (a justification is given in [5]). Thus, the ratio of expected heuristic to optimal tour lengths remains unchanged.

Other metrics.

We may take as our measure of distance (in U) the l_p norm

$$\|\xi-\xi'\|_p = \left(\sum_{i=1}^{d} (\xi_i-\xi_i')^p \right)^{1/p} \qquad p\geq 1$$

This includes as special cases the "sum of coordinates" or "rectilinear" metric (p=1) and "largest cooordinate" metric ($p=\infty$), as well as the Euclidean metric (p=2). Since $\|x\|_p / \|x\|_2$ lies between 1 and $d^{(1/p)-(1/2)}$, a curve satisfying (P2) for the Euclidean norm also satisfies (P2) for any l_p norm.

<u>Performance enhancement.</u>

Our method can be extended slightly so that it generates better tours in the square, still in $O(N \log N)$ operations. The additional step, briefly, is as follows. After computing the spacefilling heuristic tour, attempt to move each point to a new location within the tour by computing three alternative values of θ. More specifically, for each point $\xi_i = \psi(\theta_i)$, find the largest triangle of the form given by Lemma 7.5, such that ξ_i lies in the triangle, but the other given points do not (i.e., the θ values for the remaining points lie outside the interval $[i \cdot 2^{-m}, (i+1) \cdot 2^{-m}]$), project the given point onto each of the three sides of this triangle so that the projection lies just <u>outside</u> the triangle, determine the position in the tour of each projected point, and move the given point to an alternative position if the tour would be shortened by doing so (Figure 2). We have estimated the length of the tour obtained by this improved heuristic to be $\sim .88 \sqrt{NA}$, that is, ~15% over optimum, and the effort to obtain an improved tour to be 4-6 times greater than that to obtain a simple spacefilling heuristic tour.

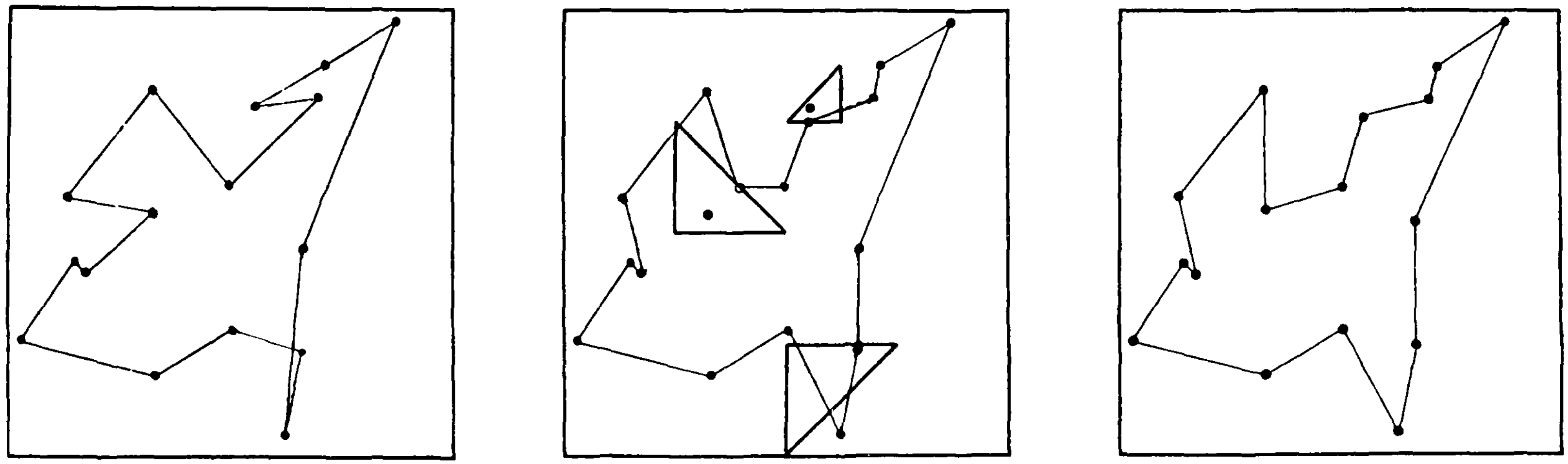


Figure 2. The optional enhancement step. The first square contains a tour of sixteen randomly generated points. About each of these, we construct a triangle of consecutive points along the spacefilling curve. It is the largest such triangle containing no other points of the tour. When a point is shifted onto the boundary of this triangle, its position in the tour may be altered, as shown in the second square. Returning the point to its original location while preserving the new sequence yields a new tour of the original points. Retaining only those sequences that reduce the tour length produces the tour shown in the third square. Some obvious inefficiencies have been eliminated. The length was reduced by 8%.

Other spacefilling curves in the plane.

Good heuristic tours can be generated from spacefilling curves other than the one given in Section 2. Hilbert's curve, for example, does not begin and end at the same point; this might be a desirable feature in some applications (such as routing a bus to start at the airport and finish downtown, then repeating the trip in reverse order). If Hilbert's curve is arranged so that it starts and ends at the same point, then our experiments show that it will yield tours assymptotically comparable to those produced by ours. For smaller N, and points distributed over a square, the tours constructed by our curve are slightly better. For smaller N and points in rectangular regions (width > 2·length), Hilbert's curve may perform better than ours.

A spacefilling curve in d-space.

A curve similar to that of Section 2 can be defined by partitioning the unit d-cube into 2^d similar subcubes. It visits the vertices of the cube (and hence, the subcubes) in a sequence determined by the Gray code [8]: (0...0), (0...0 1), (0...0 1 1), (0...0 1 0), (0...0 1 1 0), ... If we take (0...0) to be the 0-th (and 2^d-th) vertex visited, then (1...1) will be the M_d-th vertex visited, where

$$M_d = \begin{cases} (2/3)\cdot(2^d-1), & d \text{ even} \\ (2/3)\cdot(2^d-.5), & d \text{ odd} \end{cases}$$

Thus, for d>2, (1...1) will not be visited halfway along the curve. Rather, $\Psi(\alpha_d) = (1...1)$ for

$$\alpha_d = \begin{cases} (2/3)\cdot(1-2^{-d}), & d \text{ even} \\ 2/3, & d \text{ odd} \end{cases}$$

Briefly, Ψ visits the subcube containing the n-th vertex as θ ranges over the interval $[(n-\alpha_d)\cdot 2^{-d},\ (n+1-\alpha_d)\cdot 2^{-d}]$, $n=1,\ldots,2^d-1$, and $\Psi(\theta)$=the n-th vertex when $\theta=n\cdot 2^{-d}$, n even, or $\theta=(n-\alpha_d)\cdot 2^{-d}$, n odd.

The inverse image under Ψ is computed by

```
FUNCTION THETA(X_1,...,X_d)
     1) For i=1 to d, set V_i ← INT(2*X_i). Choose
        N∈{1,...,2^d} so that (V_1...V_d) is the N-th
        vertex visited by the curve.
     2) For i=1 to d, set Y_i ← 1 - 2*min(X_i,1-X_i).
        Compute θ ← THETA(Y_1,...,Y_d).
        If N is odd, then set θ ← 1 - θ.
        RETURN((N+θ-α_d)*2^-d mod 1).
```

Thus, it is clear that (P1) is satisfied.

To show that (P2) holds (with $\|\cdot\|$ = Euclidean distance), let θ_1 and θ_2 be any two points in C, and set $\Delta = |\theta_1-\theta_2|$, $D = \|\psi(\theta_1)-\psi(\theta_2)\|$. Suppose that the cube is broken into 2^{kd} subcubes of side 2^{-k}. We call any two subcubes neighbors if they have at least one vertex in common. (By this definition, a subcube is its own neighbor.) Our curve visits each subcube twice; each visit corresponds to an interval in C of length $\alpha_d\ 2^{-kd}$ or $(1-\alpha_d)\ 2^{-kd}$; and consecutive intervals in C are always mapped into neighboring subcubes. Thus, if $\Delta < \min(\alpha_d, 1-\alpha_d)\ 2^{-kd}$, then $\psi(\theta_1)$ and $\psi(\theta_2)$ lie in neighboring subcubes, and so $D \leq 2\ d^{1/2}\ 2^{-k}$. (Note that this argument is valid only for integer k.) Thus, (P2) holds with $\Omega = 4\ d^{1/2}\ (1-\alpha_d)^{-1/d}$. (We note that the Ω obtained in this manner is not as good as the Ω established in Theorem 3.1 by an argument that is valid only for d=2.) Consequently, the spacefilling heuristic will produce a tour of length at most $O(d^{1/2}\ N^{(d-1)/d})$. Beardwood, Halton, and Hammersley[5] have shown that the average optimal tour is of this order of magnitude.

6. Proof of Theorem 2.2. The proof is based upon the following identity: if Λ is a set of nonnegative numbers and $H(t) = \#\{\ \lambda \epsilon \Lambda \mid \lambda \geq t\ \}$, then the sum of elements in Λ equals $\int_0^\infty H(t)\, dt$. We let Λ represent the set of N link lengths along the heuristic tour and construct a bound of the form $H^+(t) \geq H(t)$, so that

$$L = \int_0^\infty H(t)\, dt \leq \int_0^\infty H^+(t)\, dt.$$

Let Ξ denote the set of N given points, and let $\Xi(\epsilon)$ denote the set of points in U that lie within distance ϵ of a point in Ξ. We first show that

$$V[\Xi(\epsilon)] \leq c' L^* \epsilon^{d-1} + c\, \epsilon^d \tag{6.1}$$

where $V[\cdot]$ denotes d-volume, $c = V[\{\ \xi \mid \|\xi\| \leq 1\ \}]$ and $c' = \sup_{\|\zeta\| \leq 1} V[\{\ \xi = \xi' + a\zeta \mid \|\xi'\| \leq 1,\ \Sigma_i \xi'_i \zeta_i = 0,\ 0 \leq a \leq 1\ \}]$. (If $\|\cdot\|$ is Euclidean distance, then c is the volume of a unit d-sphere and c' is the volume of a unit d-cylinder. Note, however, that $\|\cdot\|$ can be any norm on d-space!) To simplify notation, let "node" indicate a point in Ξ, and let "link" indicate a segment of the optimal tour of Ξ. A point will be considered to "lie within ϵ of a link" if it lies within distance ϵ of some point along the link. Since any given set of i nodes (i=1,...,N-1) adjoins at least i+1 links,

$$\{\xi \mid \xi \text{ lies within } \epsilon \text{ of } i \text{ or more nodes}\}$$
$$\subseteq \ \{\xi \mid \xi \text{ lies within } \epsilon \text{ of } i{+}1 \text{ or more links}\},\ i=1,\dots,N-1$$

It follows that

$$V[\{\xi \mid \xi \text{ lies within } \epsilon \text{ of } 1 \text{ or more nodes}\}] + \Sigma_{i=1}^{N} V[\{\xi \mid \xi \text{ lies within } \epsilon \text{ of } i \text{ or more nodes}\}]$$

$$\le \Sigma_{i=1}^{N} V[\{\xi \mid \xi \text{ lies within } \epsilon \text{ of } i \text{ or more links}\}] + V[\{\xi \mid \xi \text{ lies within } \epsilon \text{ of all } N \text{ nodes}\}] \qquad (6.1')$$

We may obtain (6.1) from (6.1') by the three substitutions given below:

$$\Sigma_{i=1}^{N} V[\{\xi \mid \xi \text{ lies within } \epsilon \text{ of } i \text{ or more nodes}\}]$$

$$= \Sigma_{i=1}^{N} i \cdot V[\{\xi \mid \xi \text{ lies within } \epsilon \text{ of } i \text{ nodes exactly}\}]$$

$$= \Sigma_{i=1}^{N} V[\{\xi \mid \xi \text{ lies within } \epsilon \text{ of the } i\text{-th node}\}]$$

$$= N\, c\, \epsilon^{d}$$

$$\Sigma_{i=1}^{N} V[\{\xi \mid \xi \text{ lies within } \epsilon \text{ of } i \text{ or more links}\}]$$

$$= \Sigma_{i=1}^{N} i \cdot V[\{\xi \mid \xi \text{ lies within } \epsilon \text{ of } i \text{ links exactly}\}]$$

$$= \Sigma_{i=1}^{N} V[\{\xi \mid \xi \text{ lies within } \epsilon \text{ of the i-th link}\}]$$

$$= c' L^{*} \epsilon^{(d-1)} + N c \epsilon^{d}$$

$$V[\{\xi \mid \xi \text{ lies within } \epsilon \text{ of all N nodes}\}] \leq c \epsilon^{d}$$

Let us partition C into k^d intervals of equal length,

$$E_i^k = \{ \theta \mid i \cdot k^{-d} \leq \theta < (i+1) \cdot k^{-d} \}, \quad i=0,1, \ldots , k^d-1$$

If the distance between any two consecutive points along the heuristic tour is $\geq f(k^{-d})$, then, by (P2), these points cannot lie within the same E_i^k, so

$$\#\{\lambda \epsilon \Lambda \mid v \geq f(k^{-d})\} \leq \#\{i \mid (\psi(E_i^k) \cap S) \text{ is nonempty}\} \quad (6.2)$$

But the points in any E_i^k lie within $f(k^{-2})$ of each other, so

$$(\psi(E_i^k) \cap \Xi) \text{ is nonempty } \Rightarrow \psi(E_i^k) \subseteq \Xi(f(k^{-d})). \tag{6.3}$$

Finally, each E_i^k has length k^{-d}, and ψ is measure preserving, so

$$V[\psi(E_i^k)] = k^{-d} \tag{6.4}$$

Combining (6.1) through (6.4), and setting $\epsilon = f(k^{-d}) = \Omega/k$, we obtain

$$\#(\lambda\epsilon\Lambda \mid \lambda \geq \Omega/k) \leq k^d V[\Xi(\Omega/k)] = c' \Omega^{d-1} k L^* + c \Omega^d$$

If $\tilde{k}$ is a nonnegative number (not necessarily integer), then

$$\#(\lambda\epsilon\Lambda \mid \lambda \geq \Omega/\tilde{k}) \leq c' \Omega^{d-1} \lceil \tilde{k} \rceil L^* + c \Omega^d$$

$$\leq c' \Omega^{d-1} (\tilde{k}+1) L^* + c \Omega^d$$

Letting $t = \Omega/\tilde{k}$, this becomes

$$\#(\lambda\epsilon\Lambda \mid \lambda \geq t) \leq \Omega^d (c' L^*/t + c' L^*/\Omega + c)$$

There are N points in S, so

$$\#(\lambda \in \Lambda \mid \lambda \geq t) \leq N.$$

The distance between any two points in S is bounded above by f(1/2) (since 1/2 maximizes f(·)) and by $L^*/2$ (because the triangle inequality holds and an optimal tour of length L^* joins the points twice). Thus

$$\#(\lambda \in \Lambda \mid \lambda \geq t) = 0, \quad t > \min(\Omega\, 2^{-d}, L^*/2).$$

These combine to form

$$L \leq \int_0^{\min(\Omega\, 2^{-d},\, L^*/2)} \min\left(N, \Omega^d\, (c'L^*/t + c'L^*/\Omega + c)\right) dt$$

So

$$L/L^* \leq \int_0^{L^*/2} \min(N/L^*, c'\Omega^d/t)\, dt + c'(\Omega/2)^d + c\Omega^d/2$$

$$= c'\, \Omega^d \int_0^{1/2\Omega^d c'} \min(N, 1/\tau)\, d\tau + \text{constant terms}$$

$$(\tau = t/c'\Omega^d L^*)$$

$$= c'\, \Omega^d \ln(N) + \text{constant terms.} \quad \square \qquad (6.5)$$

7. Proof of Theorem 3.1. We consider first a spacefilling curve closely related to ψ, and defined by a similar recursive structure in which the domain is broken into two, rather than four, identical parts. This related spacefilling function will be denoted by $\hat{\psi}$; its domain is the set

$$D = \{ \omega \mid 0 \leq \omega \leq 2 \}; \tag{7.1}$$

and its range is the triangle

$$T = \{ (x,y) \mid x+y \leq 2, \; x,y \geq 0 \}. \tag{7.2}$$

As ω ranges from 0 to 2, the related spacefilling curve $\hat{\psi}(\omega)$ will continuously cover every point in T, starting at the lower right vertex (2,0) and ending at the upper left vertex (0,2).

We define $\hat{\psi}$ recursively, by splitting T into two smaller triangles, each filled with a spacefilling curve (Figure 3). The smaller triangles are given by

$$\begin{aligned} T_0 &= \{ (x,y) \in T \mid x \geq y \} \\ T_1 &= \{ (x,y) \in T \mid x \leq y \} \end{aligned} \tag{7.3}$$

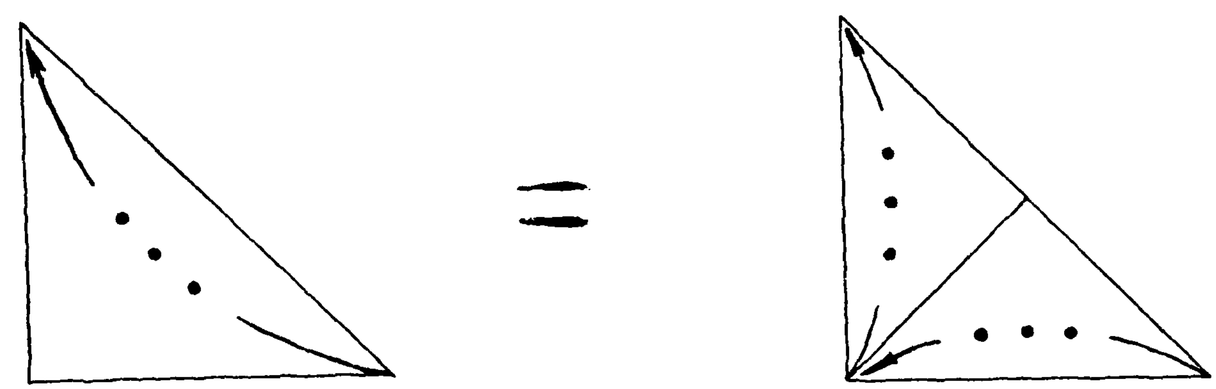


Figure 3. *Structure of the spacefilling curve in the triangle.*

We also divide the domain D into two halves

$$D_0 = \{ \omega \in D \mid \omega \le 1 \}$$
$$D_1 = \{ \omega \in D \mid \omega \ge 1 \} \qquad (7.4)$$

so that $\hat{\Psi}$ maps each D_i onto T_i. Finally, we construct affine mappings A_i from T to T_i, and b_i from D_i to D, each completely determined by the following information

$$A_0 : (2,0) \to (2,0),\ (0,0) \to (1,1),\ (0,2) \to (0,0)$$
$$A_1 : (2,0) \to (0,0),\ (0,0) \to (1,1),\ (0,2) \to (0,2) \qquad (7.5)$$
$$b_0 : 0 \to 0,\ 1 \to 2 \qquad\qquad b_1 : 1 \to 0,\ 2 \to 2$$

The recursive structure of $\hat{\Psi}$ may now be written as

$$\hat{\Psi}(\omega)=A_i[\hat{\Psi}(b_i[\omega])], \qquad \omega \in D_i, \qquad i=0,1 \qquad (7.6)$$

<u>Lemma 7.1.</u> Eq. (3.6) has a unique solution ψ if and only if (7.6) has a unique solution $\hat{\psi}$. Furthermore, if both solutions exist, then

$$\psi(\theta) = (1,1) - \hat{\psi}(\theta+.5), \quad \theta \epsilon C \tag{7.7}$$

<u>Proof.</u> Let $\theta = (j+h)/8$, where j is an integer, and $0 \leq h < 1$. Each case $j=0,1,\ldots,7$ must be considered separately. We sketch the case $j=0$ only. Suppose first that $\hat{\psi}$ exists and (7.7) holds. The LHS of (3.6) is $\psi(\theta)=\psi(h/8)$. By (7.7), this becomes $(1,1) - \hat{\psi}(.5+h/8)$. But the argument of $\hat{\psi}$ is less than one, so (7.6) permits this expression to be converted to $(1,1) - A_0[\hat{\psi}(1+h/4)]$. Continuing in this manner, the LHS of (3.6) is seen to be $(1,1) - A_0[A_1[A_0[A_0[\hat{\psi}(2h)]]]]$. The RHS of (3.6) becomes $R_0[(1,1)-A_1[A_0[\hat{\psi}(2h)]]]$. These expressions are equivalent. Similarly, if we assume that ψ exists, we can construct $\hat{\psi}$. □

Eq. (7.6) may also be written as a fixed-point identity

$$\hat{\psi} = F\,\hat{\psi} \tag{7.8}$$

where F is a transformation on the space of T-valued functions on D. For any function $\phi : D \to T$, the image of ϕ under F is given by

$$[F\Phi](\omega) = A_i[\Phi(b_i[\omega])], \qquad \omega\in D_i, \qquad i=0,1 \qquad (7.9)$$

The operator F provides a basis for demonstrating the existence of $\hat{\Psi}$.

<u>Lemma 7.2.</u> The transformations A_i and b_i are distance preserving, in the following sense:

a) $\| A_i[(x,y)] - A_i[(x',y')]\|_2 \;=\; 2^{-1/2}\,\|(x,y)-(x',y')\|_2$,
for any $(x,y),(x',y')\in T$, and $i=0$ or 1.

b) $|b_i[\omega] - b_i[\omega']| = 2\,|\omega-\omega'|$,
for any $\omega,\omega'\in D_i$, and $i=0$ or 1.

<u>Corollary 7.2A.</u> F is a contraction operator under the sup (Euclidean) norm, i.e.,

$$\sup_{\omega\in D}\{\|[F\Phi](\omega)-[F\Phi'](\omega)\|_2\} \le 2^{-1/2}\sup_{\omega\in D}\{\|\Phi(\omega)-\Phi'(\omega)\|_2\}.$$

<u>Corollary 7.2B.</u> For any "initial guess" $\Phi: D\to T$, the sequence of functions $F^n\Phi$, $n=0,1,\ldots$, converges uniformly to a (unique) solution of (7.8) [equivalently, of (7.6)].

<u>Corollary 7.2C.</u> There exists a unique continuous function ψ satisfying (3.6).

<u>Proof:</u> Let $\hat{\psi}$ be the unique continuous solution of (7.8) and apply Lemma 7.1. □

A simple initial guess for $\hat{\psi}$ is

$$\phi(\omega) = (2-\omega,\ \omega), \qquad \omega \in D \tag{7.10}$$

Let $\phi^n = F^n\phi$. By Corollary 7.2B, $\phi^n \to \hat{\psi}$. Figure 4 shows the first few terms of this sequence. In view of (7.7), the restriction of ϕ^n to S (shown within dotted lines in Figure 4) is a convergent sequence of approximations to ψ, rotated by 180°.

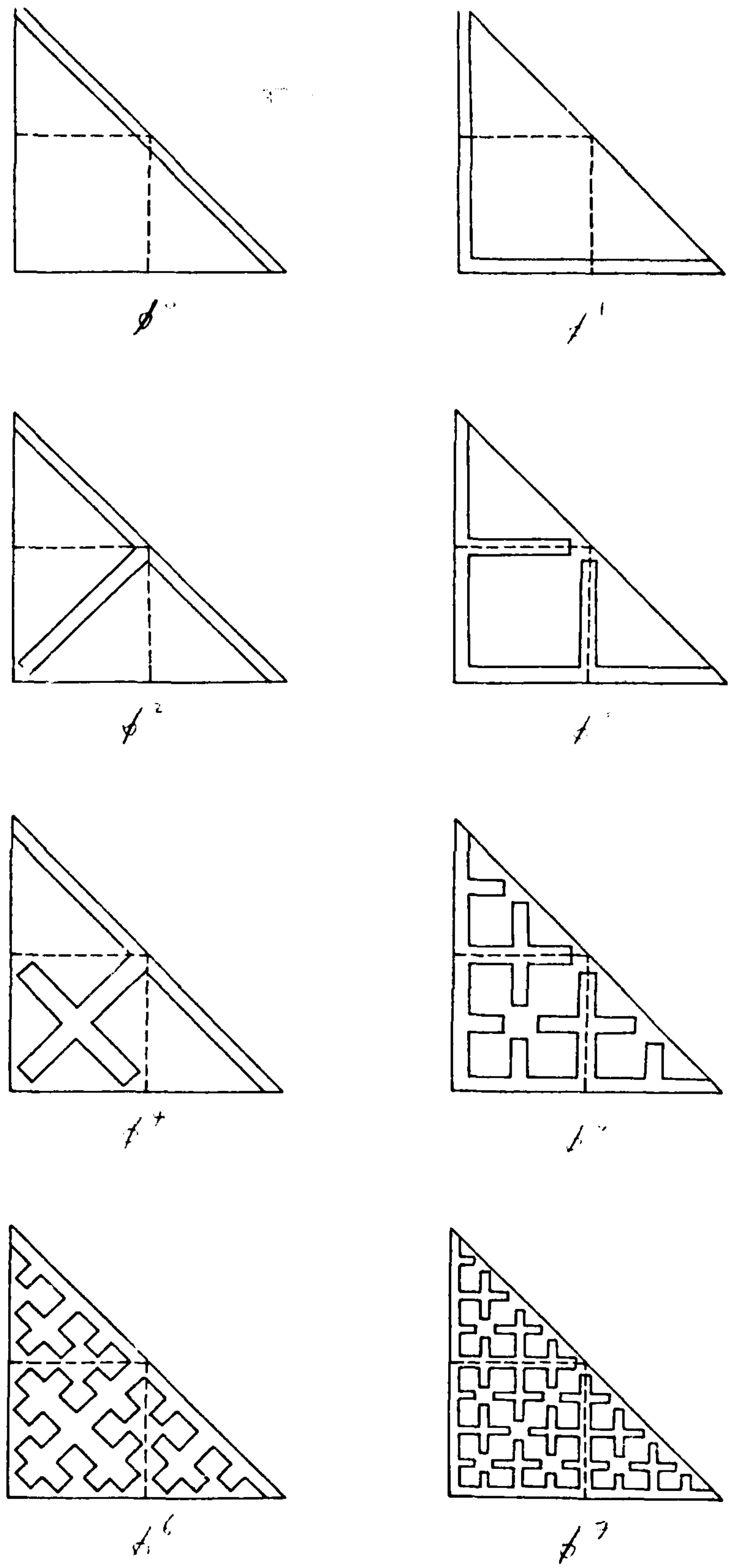


Figure 4. The first eight terms of a sequence of approximations to the spacefilling curve $\hat{\psi}$. The approximations to ψ are shown within the dotted lines.

We now turn our attention to establishing (P1).

Lemma 7.3. If X and Y are both integer multiples of 2^{-k}, then

(a) ψ(THETA(X,Y)) = (X,Y).

(b) THETA(X,Y) will be an integer multiple of $2^{-2(k+1)}$.

(c) THETA(X,Y) will call itself at most k times.

Proof: An induction on k follows easily from the structure of R_i and q_i. □

Corollary 7.3A The function ψ satisfies property (P1).

Corollary 7.3B. ψ is a spacefilling curve over S.

Proof: Let $\tilde{S}$ denote the points in S whose coordinates have finite binary expansion. THETA will compute, for $(x,y)\in\tilde{S}$, a θ such that $\psi(\theta) = (x,y)$. Consequently, $\tilde{S} \subseteq \psi(C) \subseteq S$. C is a compact set, so the continuity of ψ (assured by Corollary 7.2C) implies that $\psi(C)$ is compact. That is, closure($\tilde{S}$) $\subseteq \psi(C)$. But closure($\tilde{S}$) = S, so $\psi(C) = S$. Since ψ is continuous and surjective, it is spacefilling. □

Next, we establish (P2).

<u>Lemma 7.4.</u> $\|\hat{\Psi}(\omega)-\hat{\Psi}(\omega')\|_2 \leq 2\sqrt{|\omega-\omega'|}, \quad \omega,\omega'\in D.$

<u>Proof.</u> Since $\Phi^n\to\hat{\Psi}$, it suffices to show that

$$\|\Phi^n(\omega)-\Phi^n(\omega')\|_2 \leq 2\sqrt{|\omega-\omega'|}, \quad \omega,\omega'\in D, \quad n=0,1,\ldots \tag{7.11}$$

This we do by induction. By (7.10),

$$\|\Phi^0(\omega)-\Phi^0(\omega')\|_2 \leq \sqrt{2}\cdot|\omega-\omega'|, \quad \omega,\omega'\in D.$$

But (7.1) ensures that $|\omega-\omega'| \leq 2$, $\omega,\omega'\in D$, and so (7.11) holds when $n=0$. To perform the induction, we consider two cases:

(a) $\omega,\omega'\leq 1$ or $\omega,\omega'\geq 1$. The induction follows trivially from Lemma 7.2.

(b) $\omega\leq 1\leq\omega'$ or $\omega'\leq 1\leq\omega$. Since any two points in T determine an angle at $(0,0)$ of, at most, 90°, the "cosine law" extension of the Pythagorean Theorem implies

$$\left(\|\phi^{n+1}(\omega)-\phi^{n+1}(\omega')\|_2\right)^2$$

$$\leq \left(\|\phi^{n+1}(\omega)-(0,0)\|_2\right)^2 + \left(\|(0,0)-\phi^{n+1}(\omega')\|_2\right)^2$$

But $\phi^{n+1}(1)=(0,0)$ [see (7.10), (7.11) and (7.5), or examine Figure 3] and so this may be written as

$$\left(\|\phi^{n+1}(\omega)-\phi^{n+1}(\omega')\|_2\right)^2$$

$$\leq \left(\|\phi^{n+1}(\omega)-\phi^{n+1}(1)\|_2\right)^2 + \left(\|\phi^{n+1}(1)-\phi^{n+1}(\omega')\|_2\right)^2$$

From (a), above, it follows that

$$\left(\|\phi^{n+1}(\omega)-\phi^{n+1}(\omega')\|_2\right)^2 \leq 4\cdot|\omega-1| + 4\cdot|1-\omega'| = 4\cdot|\omega-\omega'|$$

This establishes the induction for case (b). □

<u>Corollary 7.4A</u> The function ψ satisfies (P2) with $\|\cdot\|$ = Euclidean distance in (2.1) and Ω = 2 in (2.2).

Now we establish (P3).

<u>Lemma 7.5.</u> For any integers $m \geq 1$ and $0 \leq i < 2^m$, the set $\{ \psi(\theta) \mid i \cdot 2^{-m} \leq \theta \leq (i+1) \cdot 2^{-m} \}$ is a right triangle, of area 2^{-m}, having 45° angles at $\psi(i \cdot 2^{-m})$ and $\psi((i+1) \cdot 2^{-m})$, and a 90° angle at $\psi((i+.5) \cdot 2^{-m})$.

<u>Proof:</u> The assertion is true of $\hat{\psi}$ for m=0 by definition of T. An induction on m is based on the recursive structure shown in Figure 3: as $m \leftarrow m+1$, each triangle is bisected at its right angle to form two new triangles of equal area. In view of Lemma 1, the property extends to ψ as well. □

<u>Corollary 7.5A.</u> ψ is Lebesgue measure preserving.

Finally, we examine the sequence of means, μ_N.

Lemma 7.6. For any N, N',

$$|\mu_N - \mu_{N'}| \leq 2\Omega \; |\ln(N/N')|.$$

Proof: If a point selected at *random* is removed from a heuristic tour of N independent random points, the remaining N-1 points will be independent and sequenced so as to form a heuristic tour. By the triangle inequality, the heuristic tour of N-1 points is no longer than the heuristic tour of N points, but it is less by, at most, the lengths of the two links adjoining the removed point. By symmetry, the expected length of a link selected at random is N^{-1} times the tour length. That is,

$$0 \leq N^{1/2} \mu_N - (N-1)^{1/2} \mu_{N-1} \leq (2/N) \, N^{1/2} \mu_N$$

Dividing by $N^{1/2}$, and using $1-N^{-1} \leq (1-N^{-1})^{1/2} \leq 1$, as well as (3.5), this becomes

$$|\mu_N - \mu_{N-1}| \leq 2\Omega/N$$

For N'<N, then,

$$|\mu_N - \mu_{N'}| \leq \sum_{i=N'+1}^{N} 2\Omega/i \leq \int_{N'}^{N} 2\Omega/t \, dt \qquad \square$$

<u>Lemma 7.7.</u> There is a sequence β_N such that $\beta_{4N} = \beta_N$ for all N, and $\mu_N - \beta_N \to 0$.

<u>Proof:</u> Consider a modification of the random N point problem, where the number of points, denoted by n, is itself a random variable having Poisson distribution with mean N. The θ_i's now constitute a Poisson process on C. Let $\mu'(N) = N^{-1/2}$ · the expected length of the heuristic tour of these n points. Also let $\mu^+(N) = N^{-1/2}$ · the expected length of a heuristic tour of the n points along with the four vertices of the square (that is, n+4 points in all), and let $\mu^-(N) = N^{-1/2}$ · the expected length of all links along the heuristic tour of the n random points and four vertices excepting those links that adjoin a vertex of the square. Now $0 \leq \mu^+(N) - \mu^-(N) \leq 8\Omega/N$, since (i) the tours described by $\mu^+(N)$ and $\mu^-(N)$ differ by (at most) the length of eight links (adjoining the vertices of the square), each of which ranges over 1/N of C on average, and (ii) $f(\cdot)$ is concave, so that $E[f(\Delta)] \leq f(E[\Delta])$. The recursive structure of Ψ (shown in Figure 1) assures that $\mu^-(N) \leq \mu^-(4N)$ and $\mu^+(N) \geq \mu^+(4N)$, so $\mu^-(N \cdot 4^k)$ and $\mu^+(N \cdot 4^k)$ converge monotonically (as $k \to \infty$) to a limit (that may depend on N). Let β_N denote this limit. Clearly $\beta_{4N} = \beta_N$. Also $\mu^-(N) \leq \beta_N \leq \mu^+(N)$. And, by their definition (above), $\mu^-(N) \leq \mu'(N) \leq \mu^+(N)$. So

$$|\mu'(N) - \beta_N| \leq \mu^+(N) - \mu^-(N) \leq 8\Omega/N.$$

The relationship between $\mu'(N)$ and μ_N is

$$\mu'(N) = E[(n/N)^{1/2}\,\rho_n] = E[(n/N)^{1/2}\,\mu_n]$$

$$= E[(n/N)^{1/2}\,(\mu_n - \mu_N)] \;+\; E[(n/N)^{1/2} - 1]\,\mu_N \;+\; \mu_N$$

This, along with Lemma 7.6 and (2.5) implies

$$|\mu'(N) - \mu_N| \leq 2\Omega\, E[(n/N)^{1/2}\,|\ln(n/N)|] + \Omega\, E[|(n/N)^{1/2} - 1|].$$

When N is large, n/N is normally distributed with mean=1 and variance=1/N, and so $\mu'(N) - \mu_N \to 0$. □

<u>Corollary 7.7A.</u> There is a continuous function $\beta(\cdot)$ such that

$$\mu_N - \beta\left(4^{\log_4(N) \bmod 1}\right) \to 0.$$

<u>Proof:</u> Let $\beta\left(4^{\log_4(N) \bmod 1}\right) = \beta_N$.
Continuity of $\beta(\cdot)$ follows from Lemma 7.6. □

The various assertions of Theorem 3.1 have now been proved.

6. Conclusions. The spacefilling heuristic should prove useful in large applications because of its speed, only O(N log N) operations, an order of magnitude faster than any other TSP heuristic commonly considered. It achieves this by the surprising tactic of ignoring the $O(N^2)$ interpoint distances.

Our method is also robust, in the sense that it constructs a single tour which is good with respect to a variety of metrics and point distributions. Thus, problem parameters required by other methods need not be measured, and need not remain constant over time. Furthermore, since it is based on sorting, our heuristic has the advantage that insertions and deletions can be made in only O(log N) operations.

The simplicity of our heuristic should make it attractive for low technology applications. It requires no real multiplications or square roots, and so should execute quickly on microprocessor-based systems. In fact, a computer isn't even required. For example, a table of precomputed θ values can be used to convert (x,y) coordinates (read from a map) to numbers that are then sorted manually. With our students Lee Collins and Bill Warden, we divised just such a routing system for an Atlanta charitable organization. It delivers meals to over 200 locations which change daily. Our heuristic is at the heart of this system, and is implemented on two card files. The system is easily maintained, cost virtually nothing, yet reduced total mileage by 13% and enabled a reduction from five to four delivery vehicles. For details, see [4].

REFERENCES

[1] Abelson, H. and diSessa, A. (1982). Turtle Geometry: The Computer as a Medium for Exploring Mathematics, MIT Press, Cambridge MA

[2] Aleph Naught (pseudonym) (1971). "Space-filling curves, or How to waste time with a plotter", Software - Practice and Experience 1, 403-410

[3] Bartholdi, J. J. and Platzman, L. K. (1982). "An O(N log N) planar travelling salesman heuristic based on spacefilling curves", Operations Research Letters 1, 121-125

[4] Bartholdi, J. J., Platzman, L. K., Collins, R. L., and Warden, W. H. (1983). "A minimal technology routing system for Meals-on-Wheels", Interfaces 13 (to appear in June)

[5] Beardwood, J., Halton, J. H., and Hammersley, J. M. (1959). "The shortest path through many points", Proc. Cambridge Philos. Soc. 55, 299-327

[6] Bentley, J. L. and Saxe, J. B. (1980). "An analysis of two heuristics for the Euclidean travelling salesman problem", Proc. 18th Annual Allerton Conference on Communication, Control and Computing, 41-49

[7] Garfinkel, R. S. and Gilbert, K. C. (1978). "The bottleneck travelling salesman problem: algorithms and probabilistic analysis", J. ACM 25, 435-448

[8] Hamming, R. W. (1980). Coding and Information Theory, Prentice-Hall, Englewood Cliffs, NJ

[9] Hobson, E. W. (1950). The Theory of Functions of a Real Variable and the Theory of Fourrier's Series, Vol I, 3rd edition, Cambridge University Press and Harren Press, Washington, DC

[10] Karp, R. M. (1977). "Probabilistic analysis of partitioning algorithms for the travelling salesman problem in the plane", Math. Operations Res. 2, 209-224

[11] Knuth, D. (1973). The Art of Computer Programming, Vol 3: Sorting and Searching, Addison-Wesley, Reading, MA

[12] Papadimitriou, C. H. (1977). "The Euclidean travelling salesman problem is NP-complete", Theoret. Comput. Sci. 4, 237-244

[13] Sierpinski, W. (1912). "Sur une nouvelle courbe qui remplit toute une aire plaine", Bull. Acad. Sci. Cracovie, Série A, 462-478

[14] Steele, J. M. (1981). "Complete convergence of short paths and Karp's algorithm for the TSP", Math. Operations Res. 6, 374-378

[15] Weide, B. W. (1978). "Statistical methods in algorithm design and analysis." Ph.D. Thesis, Carnegie-Mellon University.

APPENDIX

The following BASIC code will compute a tour of N points in a square of side S. Variables whose names start with I-N will contain integer values. This program may require modifications to run under some versions of BASIC (e.g. eliminate variable dimensions in 30 and 410). Our sorting algorithm is HEAPSORT (see Knuth[11]).

The structure of the program is as follows:

Statements 10-90	input and output of variables
Statements 100-150	calculation of θ_i for (x_i, y_i)
Statements 200-330	sort θ_i's
Statements 400-820	optional improvement procedure

```
10  INPUT "NUMBER OF POINTS"; N : K=10 : KP = 2↑(K-1)
20  DIM A$(N),X(N),Y(N),TH(N),IQ(K),NR(N)
30  INPUT "SIDE OF SQUARE"; S : U=S*.501/KP
40  PRINT "NAME, X, Y:" : FOR I=1 TO N
        : INPUT A$(I),X(I),Y(I) : NEXT I
50  FOR I=1 TO N : GOSUB 100 : NR(I)=I : NEXT I
60  GOSUB 200 : GOSUB 400
70  PRINT : PRINT "RANK","NAME","X","Y","THETA"
80  FOR I=1 TO N : J=NR(I)
        : PRINT I,A$(J),X(J),Y(J),TH(J) : NEXT I
90  STOP


100 REM                         *** SUBROUTINE TO COMPUTE TH(I) ***
110 KX=INT(X(I)/U) : KY=INT(Y(I)/U)
120 FOR J=1 TO K : JX=INT(KX/KP) : JY=INT(KY/KP)
      : KX=2*(KX-KP*JX) : KY=2*(KY-KP*JY)
122   : IQ(J)=JY+3*JX-2*JX*JY : NEXT J
130 T=IQ(K)/4
140 FOR J=K-1 TO 1 STEP -1 : T=T+(6-IQ(J))/4
      : T=T-INT(T) : T=(3.5+T+IQ(J))/4 : NEXT J
150 TH(I)=T-INT(T) : RETURN


200 REM                  *** SUBROUTINE TO SORT ARRAY NR SO THAT ***
210 REM                  ***  TH(NR(1))<TH(NR(2))<...<TH(NR(N))  ***
220 IL=INT(N/2)+1 : IR=N
230 IF IL>1 THEN IL=IL-1 : NA=NR(IL) : GOTO 260
240 NA=NR(IR) : NR(IR)=NR(1) : IR=IR-1
250 IF IR=1 THEN NR(1)=NA : RETURN
260 TA=TH(NA) : J=IL
270 I=J : J=2*J : IF J=IR THEN GOTO 300
280 IF J>IR THEN GOTO 320
290 IF TH(NR(J))<TH(NR(J+1)) THEN J=J+1
300 IF TA>=TH(NR(J)) THEN GOTO 320
310 NR(I)=NR(J) : GOTO 270
320 NR(I)=NA : GOTO 230
```

```
400 REM                              *** OPTIONAL IMPROVEMENT SECTION ***
410 DIM LF(N),LB(N),D2(3),KV(3),KW(3) : KR=KP*2-1
420 NR(0)=NR(N) : FOR I=1 TO N : LF(NR(I-1))=NR(I)
      : LB(NR(I))=NR(I-1) : NEXT I : REM     *** SET UP LINKS ***
430 FOR II=1 TO N : JJ=NR(II) : MX=X(JJ)/U : MY=Y(JJ)/U
440   T=TH(JJ) : REM      ***  GET TRIANGLE CONTAINING POINT II ***
450   TB=TH(LB(JJ)) : IF TB>T THEN TB=0
460   TF=TH(LF(JJ)) : IF TF<T THEN TF=1
470   TL=-.5 : TG=1.5 : KV(1)=-KR : KW(1)=KR
        : KV(2)=KR : KW(2)=KR : KV(3)=KR : KW(3)=-KR
480   TM=(TL+TG)/2 : IF TB<TL AND TF>TG THEN GOTO 560
490   IF T>TM THEN GOTO 510
500   TG=TM : IV=3 : GOTO 520
510   TL=TM : IV=1
520   KK=(KV(1)+KV(3))/2 : KV(IV)=KV(2) : KV(2)=KK
530   KK=(KW(1)+KW(3))/2 : KW(IV)=KW(2) : KW(2)=KK
540   GOTO 480
550   REM            *** TRIANGLE VERTICES ARE (U*KV(.),U*KW(.)) ***
560   FOR IV=1 TO 3 : D2(IV)=(MX-KV(IV))↑2+(MY-KW(IV))↑2
        : NEXT IV : KV(0)=KV(3) : KW(0)=KW(3) : D2(0)=D2(3)
570   DS=1E9 : IS=0 : REM                       *** TEST EACH PROJ'N ***
580   FOR IV=1 TO 3
590     RD=((KV(IV)-KV(IV-1))↑2+(KW(IV)-KW(IV-1))↑2)
600     R1=(1-(D2(IV)-D2(IV-1))/RD)/2 : R2=1-R1
610     KX=R1*KV(IV)+R2*KV(IV-1) : X(0)=U*(KX+SGN(KX-MX))
620     KY=R1*KW(IV)+R2*KW(IV-1) : Y(0)=U*(KY+SGN(KY-MY))
630     I=0 : GOSUB 100 : REM           *** FIND THETA FOR PROJ'N ***
640     T=TH(0) : IF T>TB AND T<TF THEN 730
650     NL=1 : NG=N : TL=0 : TG=1
660     NM=INT((NL+NG)/2) : TM=TH(NR(NM)) : IF NG-NL<2 THEN 690
670     IF T>TM THEN TL=TM : NL=NM : GOTO 660
680     TG=TM : NG=NM : GOTO 660
690     JL=NR(NL) : JG=NR(NG) : REM      ***  NEW POS'N TO TEST ***
700     IF LF(JL)<>JG OR JL=JJ OR JG=JJ THEN GOTO 730
710     D=SQR((X(JL)-X(JJ))↑2+(Y(JL)-Y(JJ))↑2)
         +SQR((X(JG)-X(JJ))↑2+(Y(JG)-Y(JJ))↑2)
         -SQR((X(JG)-X(JL))↑2+(Y(JG)-Y(JL))↑2)
720     IF D<DS THEN DS=D : IS=IV : TS=T : JS=JL
730     NEXT IV
740   IF IS=0 THEN GOTO 800
750   JB=LB(JJ) : JF=LF(JJ) : REM    *** BETTER THAN OLD POS'N? ***
760   D=SQR((X(JF)-X(JJ))↑2+(Y(JF)-Y(JJ))↑2)
        +SQR((X(JB)-X(JJ))↑2+(Y(JB)-Y(JJ))↑2)
        -SQR((X(JF)-X(JB))↑2+(Y(JF)-Y(JB))↑2)
770   IF D<=DS THEN GOTO 800
780   NR(II)=LF(JJ) : TH(JJ)=TS : JH=LF(JS) : LB(JJ)=JS
790   LF(JJ)=JH : LF(JS)=JJ : LB(JH)=JJ : LF(JB)=JF : LB(JF)=JB
800   NEXT II
810 J=NR(1) : FOR I=2 TO N : J=LF(J) : NR(I)=J : NEXT I
820 RETURN
```

MED

-83